

```

import pandas as pd
import numpy as np

df=pd.read_csv('/content/ex.csv')
df.head()

df.isnull().sum()

df.dropna(inplace=True)

df.isnull().sum()

df.duplicated().sum()

df=df.drop_duplicates()

df.duplicated().sum()

df.shape

df.head()

df['User-Rating']

l=[]
for i in df['User-Rating']:
    l.append(i[:3])
l

df['User-Rating']=l
df

df['Album/Movie']=df['Album/Movie'].str.replace(",")
df['Singer/Artists']=df['Singer/Artists'].str.replace(",")
df

df['Singer/Artists']=df['Singer/Artists'].str.replace(":")
df

df['tags']=df['Singer/Artists']+" "+df['Genre']+" "+df['Album/Movie']+" "+df['User-Rating']
df['tags'][0]

new_df=df[['Song-Name','tags']]
new_df

new_df['tags']=new_df['tags'].apply(lambda x:x.lower())
new_df

from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer(max_features=2000)

vectors=cv.fit_transform(new_df['tags']).toarray()

vectors.shape

```

```
from sklearn.metrics.pairwise import cosine_similarity
similarity=cosine_similarity(vectors)

sorted(list(enumerate(similarity[0])).reverse=True,key=lambda x:x[1])

new_df.rename(columns={'Song-Name':'title'},inplace=True)

def recommend(music):
    music_index=new_df[new_df['title']==music].index[0]
    distances=similarity[music_index]
    music_list=sorted(list(enumerate(distances)).reverse=True,key=lambda x:x[1])[1:6]
    for i in music_list:
        print(new_df.iloc[i[0]].title)

recommend('Proper Patola')

df.head(50)

import pickle
pickle.dump(new_df,open('musicrec.pkl','wb'))

pickle.dump(similarity,open('similarities.pkl','wb'))
```